UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P O Box 1450
Alexandria, Virginia 22313-1450
www uspto gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/520,922	01/12/2005	Yoshiyuki Okimoto	2004 2068A	4924

513 7590 08/21/2008

WENDEROTH, LIND & PONACK, L.L.P.
2033 K STREET N. W.
SUITE 800
WASHINGTON, DC 20006-1021

EXAMINER	
COLUCCI, MICHAEL C	
ART UNIT	**PAPER NUMBER**
2626	
MAIL DATE	**DELIVERY MODE**
08/21/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No. 10/520,922	Applicant(s) OKIMOTO ET AL.	
	Examiner MICHAEL C. COLUCCI	Art Unit 2626	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on _____.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1-31 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-31 is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

Response to Arguments

1. Applicants arguments with respect to claims 1-31 have been considered but are moot in view of the new grounds of rejection.

Argument 1 (page 18 paragraphs 1-4):

- “However, the classification relied on in Pentheroudakis by the Examiner is implemented merely for a single word, not for a sentence or a series of words. Specifically, Pentheroudakis describes a technique in which, when a token or word such as "brother-in-law" is included in a sentence, "brother-in-law" is segmented as sub-tokens "brother," "in," and "law." However, during language processing the token "brother-in-law" is still treated as one word. In other words, in the case a sentence like "they are brothers-in-law," the classification described in Pentheroudakis would likely segment "they are brothers-in-law," into three tokens, such as "they," "are," and "brothers-in-law"; and each token would be treated equally”

Response to argument 1:

Examiner takes the position that although Rigazio et al. US 6182039 B1 (hereinafter Rigazio) and Pentheroudakis et al. US.7092871 B2 (hereinafter Pentheroudakis) teach language modeling and classification of text, both teachings do not teach the classification and modeling of multiple words and sentences. Ragazio and Pentheroudakis teach the recognition of more than one word but several words are linked by a hyphen (i.e. brother-in-law). When read in light of the specification, "two or more words", as taught in the present

invention is construed as two words without any conjoined punctuation (i.e. hyphen). Therefore, the reference of Bai et al. US 6311152 B1 (hereinafter Bai) has been introduced to address the limitation of "modeling each of a plurality of texts as a sequence of words that includes a word string class indicating a linguistic property of a word string constituting two or more words", wherein Bai teaches the collection and modeling of various text samples such as newspapers, books, magazines, etc. combined into a corpus, wherein two or more words can be recognized such as "John Doe" and segmented on a frame by frame basis into word boundaries to identify various properties of groups of words.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 1-5, 7-10, 13-18, 22, 23 and 26-31 rejected under 35 U.S.C. 103(a) as being unpatentable over Rigazio et al. US 6182039 B1 (hereinafter Rigazio) in view of Pentheroudakis et al. US 7092871 B2 (hereinafter Pentheroudakis) and further in view of Bai et al. US 6311152 B1 (hereinafter Bai).

Re claims 1, 13, 14, and 26-30, Rigazio teaches language model generation and accumulation apparatus that generates and accumulates language models for speech recognition, the apparatus comprising:

a lower-level N-gram language model (Col. 6 lines 11-20) generation and accumulation unit operable to generate and accumulate a lower-level N-gram language model that is obtained by modeling (Col. 4 lines 30-55 & Fig. 2) a sequence of two or more words within the word string class;

However, Rigazio fails to teach a word string class and a plurality of text as a second sequence of words that includes the word string class.

Pentheroudakis teaches a classification of words or a group of words that can represent a title, where a lexicon lookup engine 208 first accesses lexicon 212, which may illustratively be a computer readable dictionary, or simply a word list, to determine whether the tokens in the proposed segmentation are recognized by, or contained in, lexicon 212. In addition, linguistic knowledge component 206 may include morphological analyzer 210. For example, if lexicon 212 contains only uninflected word forms (i.e., lemmas), then a morphological analysis is desirable to reduce, say the token "brothers-in-law" to the dictionary form "brother-in-law." Pentheroudakis teaches a morphological analyzer 210 can also do more than simply convert words to uninflected forms. For example, morphological analyzer 210 also illustratively includes a number morphological component 216 and a punctuation morphological component 218. These two components illustratively convert numbers and punctuation characters to values which will be recognized by lexicon 212 as well. Additionally, if a sub-token is

successfully looked up in lexicon 212, and thus validated by linguistic knowledge component 206, that sub-token will not be further broken down. Instead, it is simply passed back to tokenizer engine 202 along with an indication that it has been validated (Pentheroudakis Col. 6 line 44 – Col. 7 line 14).

Pentheroudakis teaches a linguistic knowledge component 206 also illustratively invokes morphological analyzer 210 to assist in recognizing "virtual" words in the language (tokens that need to be treated as single words by the system, even though they are not listed in the dictionary). For instance, tokens such as numbers, electronic mail addresses, drive path names, URLs, emoticons, and the like, can be represented as a single word. Morphological analyzer 210 can assist in recognizing each segment as an actual word, or as a virtual word, by identifying it as a virtual word or reducing it to a normalized form for recognition in lexicon 212.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Ragazio to incorporate a word string class and a plurality of text as a second sequence of words that includes the word string class as taught by Pentheroudakis to allow for the recognition of a unique *sequence of words* that are not considered to be words them selves but can be categorized as a word themselves, wherein creating a system to recognize groups of words allows for a reduced amount of error during speech recognition, where punctuation (i.e. "-", "/", ".", etc) will be considered in a manner in which any title or name can be recognized and classified. (Pentheroudakis Col. 6 line 44 – Col. 7 line 14).

However, Ragazio in view of Pentheroudakis fail to teach a higher-level N-gram language model generation and accumulation unit operable to generate and accumulate a higher-lever N-gram language model that is obtained by modeling each of a plurality of texts as a sequence of words that includes a word string class indicating a linguistic property of a word string constituting two or more words.

Bai teaches a high level modeling method, wherein a decoder module (200 of FIG. 2) as a segmentor and segment the character string text and name corpus into word-string format. A single text corpus can be used, which preferably is a collection of materials from newspapers, magazines, books, etc. Initially, the contextual and entity models are not present and a longest matching method is applied by a statistical decoder to lattices produced by a lattice constructor. Each entity model corresponds to an entity corpus consisting of lists of names (entities). For example, a person's name model can be built from a person's name corpus. Likewise, an organization name model can be built from an organization name corpus, and so on. Then, word n-grams (e.g., uni-gram and bi-gram) are generated and the relevant statistics are calculated for the text corpus and different name corpus Additionally, Bai teaches a word-based language model is a direct and simple mechanism that calculates probabilities of word sequences by multiplying the probability of each word (word unigram model) or by multiplying of each n-1 preceding words in the word sequences. This can give rise to parameter explosion and a sparse data problem. A class-based language model overcomes' these problems. This model uses classes fewer in number to estimate the

probabilities of word sequences, rather than the words themselves, by mapping groups of words into classes according to predetermined criteria (Bai Col. 4 lines 1-49).

Bai teaches the identification of words and features of words, wherein a character string starting from the pointer is matched with the system lexicon to find all possible sub-strings/candidate words. In FIG. 11, two words are indicated in the first frame having the characters "C1"and "C1C2", respectively. In step 516 candidate words are put into the word lattice at the indexed position. Along with the candidate words, their word features (i.e., entity features) and statistical information are put into the word lattice. (Bai Col. 10 lines 21-39).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Ragazio in view of Pentheroudakis to incorporate a higher-level N-gram language model generation and accumulation unit operable to generate and accumulate a higher-lever N-gram language model that is obtained by modeling each of a plurality of texts as a sequence of words that includes a word string class indicating a linguistic property of a word string constituting two or more words as taught by Bai to allow for the extraction of multiple words within multiple texts as part of a class-based language model, wherein words are mapped and classified using probabilistic means (Bai Col. 4 lines 1-49).

Re claims 2 and 15, Rigazio teaches the language model generation and accumulation apparatus according to Claim 1, wherein the higher-level N-gram language model (Col. 6 lines 11-20) generation and accumulation unit and the lower-

level N-gram language model generation and accumulation unit generate the respective language models (Col. 4 lines 4-55 & Fig. 2), using different corpuses (Col. 7 line 21 – Col. 8 line 19).

However, Ragazio in view of Pentheroudakis fails to teach the higher-level N-gram language model of claim 1.

Bai teaches a high level modeling method, wherein a decoder module (200 of FIG. 2) as a segmentor and segment the character string text and name corpus into word-string format. A single text corpus can be used, which preferably is a collection of materials from newspapers, magazines, books, etc. Initially, the contextual and entity models are not present and a longest matching method is applied by a statistical decoder to lattices produced by a lattice constructor. Each entity model corresponds to an entity corpus consisting of lists of names (entities). For example, a person's name model can be built from a person's name corpus. Likewise, an organization name model can be built from an organization name corpus, and so on. Then, word n-grams (e.g., uni-gram and bi-gram) are generated and the relevant statistics are calculated for the text corpus and different name corpus Additionally, Bai teaches a word-based language model is a direct and simple mechanism that calculates probabilities of word sequences by multiplying the probability of each word (word unigram model) or by multiplying of each n-1 preceding words in the word sequences. This can give rise to parameter explosion and a sparse data problem. A class-based language model overcomes' these problems. This model uses classes fewer in number to estimate the

probabilities of word sequences, rather than the words themselves, by mapping groups of words into classes according to predetermined criteria (Bai Col. 4 lines 1-49).

Bai teaches the identification of words and features of words, wherein a character string starting from the pointer is matched with the system lexicon to find all possible sub-strings/candidate words. In FIG. 11, two words are indicated in the first frame having the characters "C1"and "C1C2", respectively. In step 516 candidate words are put into the word lattice at the indexed position. Along with the candidate words, their word features (i.e., entity features) and statistical information are put into the word lattice. (Bai Col. 10 lines 21-39).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Ragazio in view of Pentheroudakis to incorporate a higher-level N-gram language model generation and accumulation unit operable to generate and accumulate a higher-lever N-gram language model that is obtained by modeling each of a plurality of texts as a sequence of words that includes a word string class indicating a linguistic property of a word string constituting two or more words as taught by Bai to allow for the extraction of multiple words within multiple texts as part of a class-based language model, wherein words are mapped and classified using probabilistic means (Bai Col. 4 lines 1-49).

Re claims 3 and 16, Rigazio teaches the language model generation and accumulation apparatus according to Claim 2, wherein the lower-level N-gram language model (Col. 6 lines 11-20) generation and accumulation unit includes a corpus update

unit operable to update the corpus (Col. 12 lines 23-41) for the lower-level N-gram language model (Col. 4 lines 4-55 & Fig. 2),

the lower-level N-gram language model generation and accumulation unit updates the lower-level N-gram language model based on the updated corpus (Col. 12 lines 23-41), and generates the updated lower-level N-gram language model (Col. 4 lines 4-55 & Fig. 2).

Re claims 4 and 17, language model generation and accumulation apparatus according to Claim 1, wherein the lower-level N-gram language model (Col. 6 lines 11-20) generation and accumulation unit analyzes the first sequence of words (Col. 4 lines 4-55 & Fig. 2), and generates the lower-level N-gram language model by modeling each sequence of the one or more morphemes based on the word string class (Col. 4 lines 4-55 & Fig. 2).

However, Rigazio fails to teach within the word string class into one or more morphemes that are the smallest language units having meanings.

Pentheroudakis teaches a classification of words or a group of words that can represent a title, where a lexicon lookup engine 208 first accesses lexicon 212, which may illustratively be a computer readable dictionary, or simply a word list, to determine whether the tokens in the proposed segmentation are recognized by, or contained in, lexicon 212. In addition, linguistic knowledge component 206 may include morphological analyzer 210. For example, if lexicon 212 contains only uninflected word forms (i.e., lemmas), then a morphological analysis is desirable to reduce, say the token

"brothers-in-law" to the dictionary form "brother-in-law." Pentheroudakis teaches a morphological analyzer 210 can also do more than simply convert words to uninflected forms. For example, morphological analyzer 210 also illustratively includes a number morphological component 216 and a punctuation morphological component 218. These two components illustratively convert numbers and punctuation characters to values which will be recognized by lexicon 212 as well. Additionally, if a sub-token is successfully looked up in lexicon 212, and thus validated by linguistic knowledge component 206, that sub-token will not be further broken down. Instead, it is simply passed back to tokenizer engine 202 along with an indication that it has been validated (Pentheroudakis Col. 6 line 44 – Col. 7 line 14).

Pentheroudakis teaches a linguistic knowledge component 206 also illustratively invokes morphological analyzer 210 to assist in recognizing "virtual" words in the language (tokens that need to be treated as single words by the system, even though they are not listed in the dictionary). For instance, tokens such as numbers, electronic mail addresses, drive path names, URLs, emoticons, and the like, can be represented as a single word. Morphological analyzer 210 can assist in recognizing each segment as an actual word, or as a virtual word, by identifying it as a virtual word or reducing it to a normalized form for recognition in lexicon 212.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Ragazio to incorporate a word string class that is analyzed into morphemes having the smallest language unit meaning as taught by Pentheroudakis to allow for a diverse recognition of data, where punctuation can be

taken into account that links several letter/words to form a group of words and modeling a group of words to represent one word allows for the recognition of a unique *sequence of words* that are not considered to be words them selves but can be categorized as a word themselves, wherein creating a system to recognize groups of words allows for a reduced amount of error during speech recognition, where punctuation (i.e. "-", "/", ".", etc) will be considered in a manner in which any title or name can be recognized and classified (Pentheroudakis Col. 6 line 44 – Col. 7 line 14).

Re claims 5 and 18, language model generation and accumulation apparatus according to Claim 1, wherein the higher-level N-gram language model (Col. 6 lines 11-20) generation and accumulation unit, and then generates the higher-level N-gram language model by modeling (Col. 4 lines 30-55 $ Fig. 2)

a sequence made up of the virtual word and the other words (Pentheroudakis Col. 6 line 44 – Col. 7 line 14),

substitutes the word string class with a virtual word (Pentheroudakis Col. 6 line 44 – Col. 7 line 14).

the word string class being included in each of the plurality of texts analyzed into morphemes (Pentheroudakis Col. 6 line 44 – Col. 7 line 14).

Pentheroudakis teaches a classification of words or a group of words that can represent a title, where a lexicon lookup engine 208 first accesses lexicon 212, which may illustratively be a computer readable dictionary, or simply a word list, to determine whether the tokens in the proposed segmentation are recognized by, or contained in,

lexicon 212. In addition, linguistic knowledge component 206 may include morphological analyzer 210. For example, if lexicon 212 contains only uninflected word forms (i.e., lemmas), then a morphological analysis is desirable to reduce, say the token "brothers-in-law" to the dictionary form "brother-in-law." Pentheroudakis teaches a morphological analyzer 210 can also do more than simply convert words to uninflected forms. For example, morphological analyzer 210 also illustratively includes a number morphological component 216 and a punctuation morphological component 218. These two components illustratively convert numbers and punctuation characters to values which will be recognized by lexicon 212 as well. Additionally, if a sub-token is successfully looked up in lexicon 212, and thus validated by linguistic knowledge component 206, that sub-token will not be further broken down. Instead, it is simply passed back to tokenizer engine 202 along with an indication that it has been validated.

Pentheroudakis teaches a linguistic knowledge component 206 also illustratively invokes morphological analyzer 210 to assist in recognizing "virtual" words in the language (tokens that need to be treated as single words by the system, even though they are not listed in the dictionary). For instance, tokens such as numbers, electronic mail addresses, drive path names, URLs, emoticons, and the like, can be represented as a single word. Morphological analyzer 210 can assist in recognizing each segment as an actual word, or as a virtual word, by identifying it as a virtual word or reducing it to a normalized form for recognition in lexicon 212.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Ragazio to incorporate a word string class

that is analyzed into morphemes having the smallest language unit meaning and the substitution of a word string class with a virtual word as taught by Pentheroudakis to allow for a diverse recognition of data, where punctuation can be taken into account that links several letter/words to form a group of words and modeling a group of words to represent one word allows for the recognition of a unique *sequence of words* that are not considered to be words them selves but can be categorized as a word themselves, wherein creating a system to recognize groups of words allows for a reduced amount of error during speech recognition, where punctuation (i.e. "-", "/", ".", etc) will be considered in a manner in which any title or name can be recognized and classified (Pentheroudakis Col. 6 line 44 – Col. 7 line 14).

Re claims 7, 9, and 22, Rigazio teaches the language model generation and accumulation apparatus according to Claim 1, further comprising

a syntactic tree generation unit operable to perform morphemic analysis as well as syntactic analysis of a text (Col. 5 lines 42-63), and generate a syntactic tree in which said-the text is structured by a plurality of layers, focusing on a node that is on said the syntactic tree (Col. 5 lines 42-63) and that has been selected on the basis of a predetermined criterion (Col. 4 lines 4-55 & Fig. 2),

wherein the higher-level N-gram language model (Col. 6 lines 11-20) generation and accumulation unit generates the higher-level N-gram language model for syntactic tree, using a first subtree (Col. 5 lines 42-63 & Fig. 4) that constitutes an upper layer from the focused node (Col. 4 lines 4-55 & Fig. 2), and

the lower-level N-gram language model (Col. 6 lines 11-20) generation and accumulation unit generates the lower-level N-gram language model for syntactic tree, using a second subtree (Col. 5 lines 42-63 & Fig. 4) that constitutes a lower layer from the focused node (Col. 4 lines 4-55 & Fig. 2)

However, Rigazio fails to teach a morphemic analysis (Pentheroudakis Col. 6 line 44 – Col. 7 line 14).

Pentheroudakis teaches a classification of words or a group of words that can represent a title, where a lexicon lookup engine 208 first accesses lexicon 212, which may illustratively be a computer readable dictionary, or simply a word list, to determine whether the tokens in the proposed segmentation are recognized by, or contained in, lexicon 212. In addition, linguistic knowledge component 206 may include morphological analyzer 210. For example, if lexicon 212 contains only uninflected word forms (i.e., lemmas), then a morphological analysis is desirable to reduce, say the token "brothers-in-law" to the dictionary form "brother-in-law." Pentheroudakis teaches a morphological analyzer 210 can also do more than simply convert words to uninflected forms. For example, morphological analyzer 210 also illustratively includes a number morphological component 216 and a punctuation morphological component 218. These two components illustratively convert numbers and punctuation characters to values which will be recognized by lexicon 212 as well. Additionally, if a sub-token is successfully looked up in lexicon 212, and thus validated by linguistic knowledge component 206, that sub-token will not be further broken down. Instead, it is simply passed back to tokenizer engine 202 along with an indication that it has been validated.

Pentheroudakis teaches a linguistic knowledge component 206 also illustratively invokes morphological analyzer 210 to assist in recognizing "virtual" words in the language (tokens that need to be treated as single words by the system, even though they are not listed in the dictionary). For instance, tokens such as numbers, electronic mail addresses, drive path names, URLs, emoticons, and the like, can be represented as a single word. Morphological analyzer 210 can assist in recognizing each segment as an actual word, or as a virtual word, by identifying it as a virtual word or reducing it to a normalized form for recognition in lexicon 212.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Ragazio to incorporate a word string class that is analyzed into morphemes having the smallest language unit meaning and the substitution of a word string class with a virtual word as taught by Pentheroudakis to allow for a diverse recognition of data, where punctuation can be taken into account that links several letter/words to form a group of words and modeling a group of words to represent one word allows for the recognition of a unique *sequence of words* that are not considered to be words them selves but can be categorized as a word themselves, wherein creating a system to recognize groups of words allows for a reduced amount of error during speech recognition, where punctuation (i.e. "-", "/", ".", etc) will be considered in a manner in which any title or name can be recognized and classified (Pentheroudakis Col. 6 line 44 – Col. 7 line 14).

However, Ragazio in view of Pentheroudakis fails to teach the higher-level N-gram language model of claim 1.

Bai teaches a high level modeling method, wherein a decoder module (200 of FIG. 2) as a segmentor and segment the character string text and name corpus into word-string format. A single text corpus can be used, which preferably is a collection of materials from newspapers, magazines, books, etc. Initially, the contextual and entity models are not present and a longest matching method is applied by a statistical decoder to lattices produced by a lattice constructor. Each entity model corresponds to an entity corpus consisting of lists of names (entities). For example, a person's name model can be built from a person's name corpus. Likewise, an organization name model can be built from an organization name corpus, and so on. Then, word n-grams (e.g., uni-gram and bi-gram) are generated and the relevant statistics are calculated for the text corpus and different name corpus Additionally, Bai teaches a word-based language model is a direct and simple mechanism that calculates probabilities of word sequences by multiplying the probability of each word (word unigram model) or by multiplying of each n-1 preceding words in the word sequences. This can give rise to parameter explosion and a sparse data problem. A class-based language model overcomes' these problems. This model uses classes fewer in number to estimate the probabilities of word sequences, rather than the words themselves, by mapping groups of words into classes according to predetermined criteria (Bai Col. 4 lines 1-49).

Bai teaches the identification of words and features of words, wherein a character string starting from the pointer is matched with the system lexicon to find all

possible sub-strings/candidate words. In FIG. 11, two words are indicated in the first frame having the characters "C1"and "C1C2", respectively. In step 516 candidate words are put into the word lattice at the indexed position. Along with the candidate words, their word features (i.e., entity features) and statistical information are put into the word lattice. (Bai Col. 10 lines 21-39).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Ragazio in view of Pentheroudakis to incorporate a higher-level N-gram language model generation and accumulation unit operable to generate and accumulate a higher-lever N-gram language model that is obtained by modeling each of a plurality of texts as a sequence of words that includes a word string class indicating a linguistic property of a word string constituting two or more words as taught by Bai to allow for the extraction of multiple words within multiple texts as part of a class-based language model, wherein words are mapped and classified using probabilistic means (Bai Col. 4 lines 1-49).

Re claims 8, 10, and 23, Rigazio teaches the language model (Col. 6 lines 11-20) generation and accumulation apparatus according to Claim 7,

wherein the lower-level N-gram language model (Col. 6 lines 11-20) generation and accumulation unit includes

a language model generation exception word judgment unit operable to judge a specific word appearing in the second subtree (Col. 5 lines 42-63)as an exception word based on a predetermined linguistic property (Col. 4 lines 30-55 $ Fig. 2), the exception

word being a word not being included as a constituent word of any subtree (Col. 4 lines 30-55 $ Fig. 2),

the lower-level N-gram language model generation and accumulation unit generates the lower-level N-gram language model (Col. 4 lines 30-55 $ Fig. 2)by dividing the exception word into (i) a syllable that is a basic phonetic unit constituting a pronunciation of the word (Col. 4 lines 30-55 $ Fig. 2) and (ii) a unit that is obtained by combining syllables, and then by modeling a sequence made up of the syllable and the unit obtained by combining syllables in dependency on a location of the exception word in the syntactic tree (Col. 5 lines 42-63) and on the linguistic property of the exception word (Col. 4 lines 30-55 $ Fig. 2)

Re claim 31, Ragazio teaches the language model generation and accumulation apparatus according to claim 1,

wherein the lower-level N-gram language model (Col. 6 lines 11-20) generation and accumulation unit is operable to represent a first sequence of words having a common linguistic property (Fig. 1 features) as the word string class, to generate and to accumulate, for each word string class, the lower-level N-gram language model that is obtained by modeling the first sequence of words included in the word string class (Col. 4 lines 30-55 & Fig. 2); and

the lower-level N-gram language model generation and accumulation unit is operable to generate and accumulate, for each word string class, the first sequence of

words having the linguistic property (Fig. 1 features) indicated by the word string class (Col. 4 lines 30-55 $ Fig. 2).

However, Ragazio fails to teach each word included in the first sequence of words and each word included in the second sequence of words are respectively morphemes which are smallest linguistic units that have meaning

replace the first sequence of words modeled in the lower-level N-grams language model included in a text which is the sequence of words with a word string class corresponding to the first sequence of word (Pentheroudakis Col. 6 line 44 – Col. 7 line 14)

Pentheroudakis teaches a classification of words or a group of words that can represent a title, where a lexicon lookup engine 208 first accesses lexicon 212, which may illustratively be a computer readable dictionary, or simply a word list, to determine whether the tokens in the proposed segmentation are recognized by, or contained in, lexicon 212. In addition, linguistic knowledge component 206 may include morphological analyzer 210. For example, if lexicon 212 contains only uninflected word forms (i.e., lemmas), then a morphological analysis is desirable to reduce, say the token "brothers-in-law" to the dictionary form "brother-in-law." Pentheroudakis teaches a morphological analyzer 210 can also do more than simply convert words to uninflected forms. For example, morphological analyzer 210 also illustratively includes a number morphological component 216 and a punctuation morphological component 218. These two components illustratively convert numbers and punctuation characters to values which will be recognized by lexicon 212 as well. Additionally, if a sub-token is

successfully looked up in lexicon 212, and thus validated by linguistic knowledge component 206, that sub-token will not be further broken down. Instead, it is simply passed back to tokenizer engine 202 along with an indication that it has been validated.

Pentheroudakis teaches a linguistic knowledge component 206 also illustratively invokes morphological analyzer 210 to assist in recognizing "virtual" words in the language (tokens that need to be treated as single words by the system, even though they are not listed in the dictionary). For instance, tokens such as numbers, electronic mail addresses, drive path names, URLs, emoticons, and the like, can be represented as a single word. Morphological analyzer 210 can assist in recognizing each segment as an actual word, or as a virtual word, by identifying it as a virtual word or reducing it to a normalized form for recognition in lexicon 212.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Ragazio to incorporate each word included in the first sequence of words and each word included in the second sequence of words are respectively morphemes which are smallest linguistic units that have meaning, wherein of the first sequence of words is replaced and modeled in the lower-level N-grams language model included in a text which is the sequence of words with a word string class corresponding to the first sequence of word as taught by Pentheroudakis to allow for a diverse recognition of data, where punctuation can be taken into account that links several letter/words to form a group of words and modeling a group of words to represent one word allows for the recognition of a unique *sequence of words* that are not considered to be words them selves but can be categorized as a word themselves,

wherein creating a system to recognize groups of words allows for a reduced amount of error during speech recognition, where punctuation (i.e. "-", "/", ".", etc) will be considered in a manner in which any title or name can be recognized and classified (Pentheroudakis Col. 6 line 44 – Col. 7 line 14).

However, Ragazio in view of Pentheroudakis fails to teach the higher-level N-gram language model generation and accumulation unit is operable to replace the first sequence of words modeled in the lower-level N-grams language model included in a text which is the sequence of words with a word string class corresponding to the first sequence of word, and to generate and to accumulate a higher-lever N-gram language model that is obtained by modeling the text which is the character string as a sequence of words that includes the word string class and a second sequence of words

Bai teaches a high level modeling method, wherein a decoder module (200 of FIG. 2) as a segmentor and segment the character string text and name corpus into word-string format. A single text corpus can be used, which preferably is a collection of materials from newspapers, magazines, books, etc. Initially, the contextual and entity models are not present and a longest matching method is applied by a statistical decoder to lattices produced by a lattice constructor. Each entity model corresponds to an entity corpus consisting of lists of names (entities). For example, a person's name model can be built from a person's name corpus. Likewise, an organization name model can be built from an organization name corpus, and so on. Then, word n-grams (e.g., uni-gram and bi-gram) are generated and the relevant statistics are calculated for

the text corpus and different name corpus Additionally, Bai teaches a word-based language model is a direct and simple mechanism that calculates probabilities of word sequences by multiplying the probability of each word (word unigram model) or by multiplying of each n-1 preceding words in the word sequences. This can give rise to parameter explosion and a sparse data problem. A class-based language model overcomes' these problems. This model uses classes fewer in number to estimate the probabilities of word sequences, rather than the words themselves, by mapping groups of words into classes according to predetermined criteria (Bai Col. 4 lines 1-49).

Bai teaches the identification of words and features of words, wherein a character string starting from the pointer is matched with the system lexicon to find all possible sub-strings/candidate words. In FIG. 11, two words are indicated in the first frame having the characters "C1"and "C1C2", respectively. In step 516 candidate words are put into the word lattice at the indexed position. Along with the candidate words, their word features (i.e., entity features) and statistical information are put into the word lattice. (Bai Col. 10 lines 21-39).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Ragazio in view of Pentheroudakis to incorporate replacing the first sequence of words modeled in the lower-level N-grams language model included in a text which is the sequence of words with a word string class corresponding to the first sequence of word, and to generate and to accumulate a higher-lever N-gram language model that is obtained by modeling the text which is the character string as a sequence of words that includes the word string class and a

second sequence of words as taught by Bai to allow for the extraction of multiple words within multiple texts as part of a class-based language model, wherein words are mapped and classified using probabilistic means (Bai Col. 4 lines 1-49).

4. **Claims 6, 11, 12, 19-21 rejected under 35 U.S.C. 103(a) as being unpatentable over Rigazio et al. US 6182039 B1 (hereinafter Rigazio) in view of Pentheroudakis et al. US 7092871 B2 (hereinafter Pentheroudakis) further in view of Bai et al. US 6311152 B1 (hereinafter Bai) and Bakis et al. US 6023673 A (hereinafter Bakis).**

Re claims 6 and 19, Rigazio teaches the language model generation and accumulation apparatus according to Claim 1,

wherein the lower-level N-gram language model (Col. 6 lines 11-20) generation and accumulation unit includes an exception word judgment unit operable to judge whether or not a specific word out of a plurality of words that appear in the word string class should be treated as an exception word (Col. 4 lines 4-55 & Fig. 2), based on a linguistic property of the specific word, and divides the exception word into (i) a syllable that is a basic phonetic unit constituting a pronunciation of the exception word (Col. 4 lines 4-55 & Fig. 2) and (ii) a unit that is obtained by combining syllables based on a judgment result the exception word being (Col. 4 lines 4-55 & Fig. 2),

the language model generation and accumulation apparatus further comprises a class dependent syllable N-gram generation and accumulation unit operable to generate class dependent syllable N-grams by modeling a sequence made up of the

syllable and the unit obtained by combining syllables and by providing a language likelihood (Col. 1 lines 31-39) to the sequence in dependency on either the word string class or the linguistic property of the exception word (Col. 4 lines 4-55 & Fig. 2),

However, Rigazio fails to teach a word not being included as a constituent word of the word string class (Pentheroudakis Col. 6 line 44 – Col. 7 line 14).

accumulate the generated class dependent syllable N-grams (Pentheroudakis Col. 6 line 44 – Col. 7 line 14)

Pentheroudakis teaches a classification of words or a group of words that can represent a title, where a lexicon lookup engine 208 first accesses lexicon 212, which may illustratively be a computer readable dictionary, or simply a word list, to determine whether the tokens in the proposed segmentation are recognized by, or contained in, lexicon 212. In addition, linguistic knowledge component 206 may include morphological analyzer 210. For example, if lexicon 212 contains only uninflected word forms (i.e., lemmas), then a morphological analysis is desirable to reduce, say the token "brothers-in-law" to the dictionary form "brother-in-law." Pentheroudakis teaches a morphological analyzer 210 can also do more than simply convert words to uninflected forms. For example, morphological analyzer 210 also illustratively includes a number morphological component 216 and a punctuation morphological component 218. These two components illustratively convert numbers and punctuation characters to values which will be recognized by lexicon 212 as well. Additionally, if a sub-token is successfully looked up in lexicon 212, and thus validated by linguistic knowledge

component 206, that sub-token will not be further broken down. Instead, it is simply passed back to tokenizer engine 202 along with an indication that it has been validated.

Pentheroudakis teaches a linguistic knowledge component 206 also illustratively invokes morphological analyzer 210 to assist in recognizing "virtual" words in the language (tokens that need to be treated as single words by the system, even though they are not listed in the dictionary). For instance, tokens such as numbers, electronic mail addresses, drive path names, URLs, emoticons, and the like, can be represented as a single word. Morphological analyzer 210 can assist in recognizing each segment as an actual word, or as a virtual word, by identifying it as a virtual word or reducing it to a normalized form for recognition in lexicon 212.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Ragazio to incorporate a word string class that is analyzed into morphemes having the smallest language unit meaning and the substitution of a word string class with a virtual word as taught by Pentheroudakis to allow for a diverse recognition of data, where punctuation can be taken into account that links several letter/words to form a group of words and modeling a group of words to represent one word allows for the recognition of a unique *sequence of words* that are not considered to be words them selves but can be categorized as a word themselves, wherein creating a system to recognize groups of words allows for a reduced amount of error during speech recognition, where punctuation (i.e. "-", "/", ".", etc) will be considered in a manner in which any title or name can be recognized and classified (Pentheroudakis Col. 6 line 44 – Col. 7 line 14).

However, Ragazio in view of Pentheroudakis fails to teach the higher-level N-gram language model of claim 1.

Bai teaches a high level modeling method, wherein a decoder module (200 of FIG. 2) as a segmentor and segment the character string text and name corpus into word-string format. A single text corpus can be used, which preferably is a collection of materials from newspapers, magazines, books, etc. Initially, the contextual and entity models are not present and a longest matching method is applied by a statistical decoder to lattices produced by a lattice constructor. Each entity model corresponds to an entity corpus consisting of lists of names (entities). For example, a person's name model can be built from a person's name corpus. Likewise, an organization name model can be built from an organization name corpus, and so on. Then, word n-grams (e.g., uni-gram and bi-gram) are generated and the relevant statistics are calculated for the text corpus and different name corpus Additionally, Bai teaches a word-based language model is a direct and simple mechanism that calculates probabilities of word sequences by multiplying the probability of each word (word unigram model) or by multiplying of each n-1 preceding words in the word sequences. This can give rise to parameter explosion and a sparse data problem. A class-based language model overcomes' these problems. This model uses classes fewer in number to estimate the probabilities of word sequences, rather than the words themselves, by mapping groups of words into classes according to predetermined criteria (Bai Col. 4 lines 1-49).

Bai teaches the identification of words and features of words, wherein a character string starting from the pointer is matched with the system lexicon to find all

possible sub-strings/candidate words. In FIG. 11, two words are indicated in the first frame having the characters "C1"and "C1C2", respectively. In step 516 candidate words are put into the word lattice at the indexed position. Along with the candidate words, their word features (i.e., entity features) and statistical information are put into the word lattice. (Bai Col. 10 lines 21-39).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Ragazio in view of Pentheroudakis to incorporate a higher-level N-gram language model generation and accumulation unit operable to generate and accumulate a higher-lever N-gram language model that is obtained by modeling each of a plurality of texts as a sequence of words that includes a word string class indicating a linguistic property of a word string constituting two or more words as taught by Bai to allow for the extraction of multiple words within multiple texts as part of a class-based language model, wherein words are mapped and classified using probabilistic means (Bai Col. 4 lines 1-49).

However, Rigazio in view of Pentheroudakis and Bai fails to teach the language likelihood being a logarithm value of a probability.

Bakis teaches in order to find the best L prototypes in the target level M, likelihoods are successively calculated starting from the top level k=1. In the top level, log-likelihoods for all N.sub.1 prototypes in that level are calculated, and the results sorted. The log-likelihood is defined as the probability that the parameter values of a prototype vector signal match the feature values of a feature vector signal under

consideration. Starting with the best prototype from the sorted list, i.e., the one with the highest log-likelihood (Bakis Col. 4 line 63 – Col. 5 line 26).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Ragazio in view of Pentheroudakis and Bai to incorporate modeling text or a plurality of text to include a word string class, where a likelihood based on a logarithmic probability is calculated as taught by Bakis to allow using for the coverage of a large range of data which can be ranked when candidate matches are found, where a system that learns or is trainable can expand its models/dictionaries to a broad range through the use of a log scale (Bakis Col. 4 line 63 – Col. 5 line 26).

Re claims 11 and 12, Rigazio teaches the language model generation and accumulation apparatus according to Claim 1,

wherein the higher-level N-gram language model (Col. 6 lines 11-20) generation and accumulation unit generates the higher-level N-gram language model in which each (Col. 4 lines 30-55 $ Fig. 2)

However, Rigazio fails to teach a sequence of N words including the word string class is associated (Pentheroudakis Col. 6 line 44 – Col. 7 line 14)

Pentheroudakis teaches a classification of words or a group of words that can represent a title, where a lexicon lookup engine 208 first accesses lexicon 212, which may illustratively be a computer readable dictionary, or simply a word list, to determine whether the tokens in the proposed segmentation are recognized by, or contained in,

lexicon 212. In addition, linguistic knowledge component 206 may include morphological analyzer 210. For example, if lexicon 212 contains only uninflected word forms (i.e., lemmas), then a morphological analysis is desirable to reduce, say the token "brothers-in-law" to the dictionary form "brother-in-law." Pentheroudakis teaches a morphological analyzer 210 can also do more than simply convert words to uninflected forms. For example, morphological analyzer 210 also illustratively includes a number morphological component 216 and a punctuation morphological component 218. These two components illustratively convert numbers and punctuation characters to values which will be recognized by lexicon 212 as well. Additionally, if a sub-token is successfully looked up in lexicon 212, and thus validated by linguistic knowledge component 206, that sub-token will not be further broken down. Instead, it is simply passed back to tokenizer engine 202 along with an indication that it has been validated.

Pentheroudakis teaches a linguistic knowledge component 206 also illustratively invokes morphological analyzer 210 to assist in recognizing "virtual" words in the language (tokens that need to be treated as single words by the system, even though they are not listed in the dictionary). For instance, tokens such as numbers, electronic mail addresses, drive path names, URLs, emoticons, and the like, can be represented as a single word. Morphological analyzer 210 can assist in recognizing each segment as an actual word, or as a virtual word, by identifying it as a virtual word or reducing it to a normalized form for recognition in lexicon 212.

However, Ragazio in view of Pentheroudakis fails to teach the higher-level N-gram language model of claim 1.

Bai teaches a high level modeling method, wherein a decoder module (200 of FIG. 2) as a segmentor and segment the character string text and name corpus into word-string format. A single text corpus can be used, which preferably is a collection of materials from newspapers, magazines, books, etc. Initially, the contextual and entity models are not present and a longest matching method is applied by a statistical decoder to lattices produced by a lattice constructor. Each entity model corresponds to an entity corpus consisting of lists of names (entities). For example, a person's name model can be built from a person's name corpus. Likewise, an organization name model can be built from an organization name corpus, and so on. Then, word n-grams (e.g., uni-gram and bi-gram) are generated and the relevant statistics are calculated for the text corpus and different name corpus Additionally, Bai teaches a word-based language model is a direct and simple mechanism that calculates probabilities of word sequences by multiplying the probability of each word (word unigram model) or by multiplying of each n-1 preceding words in the word sequences. This can give rise to parameter explosion and a sparse data problem. A class-based language model overcomes' these problems. This model uses classes fewer in number to estimate the probabilities of word sequences, rather than the words themselves, by mapping groups of words into classes according to predetermined criteria (Bai Col. 4 lines 1-49).

Bai teaches the identification of words and features of words, wherein a character string starting from the pointer is matched with the system lexicon to find all possible sub-strings/candidate words. In FIG. 11, two words are indicated in the first frame having the characters "C1"and "C1C2", respectively. In step 516 candidate

words are put into the word lattice at the indexed position. Along with the candidate words, their word features (i.e., entity features) and statistical information are put into the word lattice. (Bai Col. 10 lines 21-39).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Ragazio in view of Pentheroudakis to incorporate a higher-level N-gram language model generation and accumulation unit operable to generate and accumulate a higher-lever N-gram language model that is obtained by modeling each of a plurality of texts as a sequence of words that includes a word string class indicating a linguistic property of a word string constituting two or more words as taught by Bai to allow for the extraction of multiple words within multiple texts as part of a class-based language model, wherein words are mapped and classified using probabilistic means (Bai Col. 4 lines 1-49).

However, Rigazio in view of Pentheroudakis and Bai fails to teach a probability at which said each sequence of N words (Bakis Col. 4 line 63 – Col. 5 line 26).

Bakis teaches in order to find the best L prototypes in the target level M, likelihoods are successively calculated starting from the top level k=1. In the top level, log-likelihoods for all N.sub.1 prototypes in that level are calculated, and the results sorted. The log-likelihood is defined as the probability that the parameter values of a prototype vector signal match the feature values of a feature vector signal under consideration. Starting with the best prototype from the sorted list, i.e., the one with the highest log-likelihood.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Ragazio in view of Pentheroudakis and Bai to incorporate modeling text or a plurality of text to include a word string class, where a likelihood based on a logarithmic probability is calculated as taught by Bakis to allow using for the coverage of a large range of data which can be ranked when candidate matches are found, where a system that learns or is trainable can expand its models/dictionaries to a broad range through the use of a log scale (Bakis Col. 4 line 63 – Col. 5 line 26).

Re claim 20, Rigazio teaches the language model generation and accumulation apparatus according to Claim 19, further comprising

a syntactic tree generation unit operable to perform morphemic analysis as well as syntactic analysis of a text (Col. 5 lines 42-63), and generate a syntactic tree in which said-the text is structured by a plurality of layers, focusing on a node that is on said the syntactic tree (Col. 5 lines 42-63) and that has been selected on the basis of a predetermined criterion (Col. 4 lines 4-55 & Fig. 2),

wherein the higher-level N-gram language model (Col. 6 lines 11-20) generation and accumulation unit generates the higher-level N-gram language model for syntactic tree, using a first subtree (Col. 5 lines 42-63 & Fig. 4) that constitutes an upper layer from the focused node (Col. 4 lines 4-55 & Fig. 2), and

the lower-level N-gram language model (Col. 6 lines 11-20) generation and accumulation unit generates the lower-level N-gram language model for syntactic tree,

using a second subtree (Col. 5 lines 42-63 & Fig. 4) that constitutes a lower layer from the focused node (Col. 4 lines 4-55 & Fig. 2)

the speech recognition apparatus comprises:

an acoustic processing unit operable to generate feature parameters from the speech (Col. 4 lines 30-55 $ Fig. 2);

a word comparison unit operable to compare a pronunciation of each word with each of the feature parameters (Col. 4 lines 30-55 $ Fig. 2), and generate a set of word hypotheses including an utterance segment of each word and an acoustic likelihood of each word (Col. 1 lines 31-39);

a word string hypothesis (Col. 12 lines 23-41) generation unit operable to generate a word string hypothesis from the set of word hypotheses with reference to the higher-level N-gram language model for syntactic tree (Col. 5 lines 42-63) and the lower-level N-gram language model for syntactic tree (Col. 5 lines 42-63), and generate a result of the speech recognition

However, Rigazio fails to teach a morphemic analysis.

Pentheroudakis teaches a classification of words or a group of words that can represent a title, where a lexicon lookup engine 208 first accesses lexicon 212, which may illustratively be a computer readable dictionary, or simply a word list, to determine whether the tokens in the proposed segmentation are recognized by, or contained in, lexicon 212. In addition, linguistic knowledge component 206 may include morphological analyzer 210. For example, if lexicon 212 contains only uninflected word forms (i.e., lemmas), then a morphological analysis is desirable to reduce, say the token

"brothers-in-law" to the dictionary form "brother-in-law." Pentheroudakis teaches a morphological analyzer 210 can also do more than simply convert words to uninflected forms. For example, morphological analyzer 210 also illustratively includes a number morphological component 216 and a punctuation morphological component 218. These two components illustratively convert numbers and punctuation characters to values which will be recognized by lexicon 212 as well. Additionally, if a sub-token is successfully looked up in lexicon 212, and thus validated by linguistic knowledge component 206, that sub-token will not be further broken down. Instead, it is simply passed back to tokenizer engine 202 along with an indication that it has been validated (Pentheroudakis Col. 6 line 44 – Col. 7 line 14).

Pentheroudakis teaches a linguistic knowledge component 206 also illustratively invokes morphological analyzer 210 to assist in recognizing "virtual" words in the language (tokens that need to be treated as single words by the system, even though they are not listed in the dictionary). For instance, tokens such as numbers, electronic mail addresses, drive path names, URLs, emoticons, and the like, can be represented as a single word. Morphological analyzer 210 can assist in recognizing each segment as an actual word, or as a virtual word, by identifying it as a virtual word or reducing it to a normalized form for recognition in lexicon 212.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Ragazio to incorporate a word string class that is analyzed into morphemes having the smallest language unit meaning and the substitution of a word string class with a virtual word as taught by Pentheroudakis to

allow for a diverse recognition of data, where punctuation can be taken into account that links several letter/words to form a group of words and modeling a group of words to represent one word allows for the recognition of a unique *sequence of words* that are not considered to be words them selves but can be categorized as a word themselves, wherein creating a system to recognize groups of words allows for a reduced amount of error during speech recognition, where punctuation (i.e. "-", "/", ".", etc) will be considered in a manner in which any title or name can be recognized and classified (Pentheroudakis Col. 6 line 44 – Col. 7 line 14).

However, Ragazio in view of Pentheroudakis fails to teach the higher-level N-gram language model of claim 1.

Bai teaches a high level modeling method, wherein a decoder module (200 of FIG. 2) as a segmentor and segment the character string text and name corpus into word-string format. A single text corpus can be used, which preferably is a collection of materials from newspapers, magazines, books, etc. Initially, the contextual and entity models are not present and a longest matching method is applied by a statistical decoder to lattices produced by a lattice constructor. Each entity model corresponds to an entity corpus consisting of lists of names (entities). For example, a person's name model can be built from a person's name corpus. Likewise, an organization name model can be built from an organization name corpus, and so on. Then, word n-grams (e.g., uni-gram and bi-gram) are generated and the relevant statistics are calculated for the text corpus and different name corpus Additionally, Bai teaches a word-based language model is a direct and simple mechanism that calculates probabilities of word

sequences by multiplying the probability of each word (word unigram model) or by multiplying of each n-1 preceding words in the word sequences. This can give rise to parameter explosion and a sparse data problem. A class-based language model overcomes' these problems. This model uses classes fewer in number to estimate the probabilities of word sequences, rather than the words themselves, by mapping groups of words into classes according to predetermined criteria (Bai Col. 4 lines 1-49).

Bai teaches the identification of words and features of words, wherein a character string starting from the pointer is matched with the system lexicon to find all possible sub-strings/candidate words. In FIG. 11, two words are indicated in the first frame having the characters "C1"and "C1C2", respectively. In step 516 candidate words are put into the word lattice at the indexed position. Along with the candidate words, their word features (i.e., entity features) and statistical information are put into the word lattice. (Bai Col. 10 lines 21-39).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Ragazio in view of Pentheroudakis to incorporate a higher-level N-gram language model generation and accumulation unit operable to generate and accumulate a higher-lever N-gram language model that is obtained by modeling each of a plurality of texts as a sequence of words that includes a word string class indicating a linguistic property of a word string constituting two or more words as taught by Bai to allow for the extraction of multiple words within multiple texts as part of a class-based language model, wherein words are mapped and classified using probabilistic means (Bai Col. 4 lines 1-49).

Re claim 21, Rigazio teaches the apparatus according to Claim 20,

wherein the lower-level N-gram language model (Col. 6 lines 11-20) generation and accumulation unit includes

a language model generation exception word judgment unit operable to judge a specific word appearing in the second subtree (Col. 5 lines 42-63)as an exception word based on a predetermined linguistic property (Col. 4 lines 30-55 $ Fig. 2), the exception word being a word not being included as a constituent word of any subtree (Col. 4 lines 30-55 $ Fig. 2),

the lower-level N-gram language model generation and accumulation unit generates the lower-level N-gram language model (Col. 4 lines 30-55 $ Fig. 2)by dividing the exception word into (i) a syllable that is a basic phonetic unit constituting a pronunciation of the word (Col. 4 lines 30-55 $ Fig. 2) and (ii) a unit that is obtained by combining syllables, and then by modeling a sequence made up of the syllable and the unit obtained by combining syllables in dependency on a location of the exception word in the syntactic tree (Col. 5 lines 42-63) and on the linguistic property of the exception word (Col. 4 lines 30-55 $ Fig. 2)

the word string hypothesis generation unit generates the result of the speech recognition (Col. 12 lines 23-41).

Re claims 24 and 25, Rigazio teaches the speech recognition apparatus according to Claim 14,

wherein the higher-level N-gram language model (Col. 6 lines 11-20) generation and accumulation unit generates the higher-level N-gram language model in which each sequence of N words (Col. 4 lines 30-55 $ Fig. 2)

the speech recognition apparatus comprises

a word string hypothesis generation unit operable to evaluate a word string hypothesis (Col. 12 lines 23-41).

However, Ragazio in view of Pentheroudakis fails to teach the higher-level N-gram language model of claim 1.

Bai teaches a high level modeling method, wherein a decoder module (200 of FIG. 2) as a segmentor and segment the character string text and name corpus into word-string format. A single text corpus can be used, which preferably is a collection of materials from newspapers, magazines, books, etc. Initially, the contextual and entity models are not present and a longest matching method is applied by a statistical decoder to lattices produced by a lattice constructor. Each entity model corresponds to an entity corpus consisting of lists of names (entities). For example, a person's name model can be built from a person's name corpus. Likewise, an organization name model can be built from an organization name corpus, and so on. Then, word n-grams (e.g., uni-gram and bi-gram) are generated and the relevant statistics are calculated for the text corpus and different name corpus Additionally, Bai teaches a word-based language model is a direct and simple mechanism that calculates probabilities of word sequences by multiplying the probability of each word (word unigram model) or by multiplying of each n-1 preceding words in the word sequences. This can give rise to

parameter explosion and a sparse data problem. A class-based language model overcomes' these problems. This model uses classes fewer in number to estimate the probabilities of word sequences, rather than the words themselves, by mapping groups of words into classes according to predetermined criteria (Bai Col. 4 lines 1-49).

Bai teaches the identification of words and features of words, wherein a character string starting from the pointer is matched with the system lexicon to find all possible sub-strings/candidate words. In FIG. 11, two words are indicated in the first frame having the characters "C1"and "C1C2", respectively. In step 516 candidate words are put into the word lattice at the indexed position. Along with the candidate words, their word features (i.e., entity features) and statistical information are put into the word lattice. (Bai Col. 10 lines 21-39).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Ragazio in view of Pentheroudakis to incorporate a higher-level N-gram language model generation and accumulation unit operable to generate and accumulate a higher-lever N-gram language model that is obtained by modeling each of a plurality of texts as a sequence of words that includes a word string class indicating a linguistic property of a word string constituting two or more words as taught by Bai to allow for the extraction of multiple words within multiple texts as part of a class-based language model, wherein words are mapped and classified using probabilistic means (Bai Col. 4 lines 1-49).

However, Rigazio in view of Pentheroudakis and Bai fail to teach including the word string class is associated with a probability at which the each sequence of words occurs (Bakis Col. 4 line 63 – Col. 5 line 26),

multiplying each probability at which the each sequence of N words including the word string class occurs (Bakis Col. 4 line 63 – Col. 5 line 26).

Bakis teaches in order to find the best L prototypes in the target level M, likelihoods are successively calculated starting from the top level k=1. In the top level, log-likelihoods for all N.sub.1 prototypes in that level are calculated, and the results sorted. The log-likelihood is defined as the probability that the parameter values of a prototype vector signal match the feature values of a feature vector signal under consideration. Starting with the best prototype from the sorted list, i.e., the one with the highest log-likelihood.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Ragazio in view of Pentheroudakis and Bai to incorporate modeling text or a plurality of text to include a word string class, where a likelihood based on a logarithmic probability is calculated as taught by Bakis to allow using for the coverage of a large range of data which can be ranked when candidate matches are found, where a system that learns or is trainable can expand its models/dictionaries to a broad range through the use of a log scale (Bakis Col. 4 line 63 – Col. 5 line 26).

Conclusion

5. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Michael C. Colucci whose telephone number is (571)-270-1847. The examiner can normally be reached on 9:30 am - 6:00 pm, Monday-Friday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Richemond Dorvil can be reached on (571)-272-7602. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Michael C Colucci/
Examiner, Art Unit 2626
Patent Examiner
AU 2626
(571)-270-1847
Michael.Colucci@uspto.gov

/Vijay B. Chawan/
for Richemond Dorvil, SPE of Art Unit 2626